UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/715,546	11/19/2003	Kenichiro Yoshii	245650US2SRD	5590

22850 7590 10/01/2008

OBLON, SPIVAK, MCCLELLAND MAIER & NEUSTADT, P.C.
1940 DUKE STREET
ALEXANDRIA, VA 22314

EXAMINER	
TANG, KENNETH	

ART UNIT	PAPER NUMBER
2195	

NOTIFICATION DATE	DELIVERY MODE
10/01/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patentdocket@oblon.com
oblonpat@oblon.com
jgardner@oblon.com

Office Action Summary	Application No.	Applicant(s)	
	10/715,546	YOSHII ET AL.	
	Examiner	Art Unit	
	KENNETH TANG	2195	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 November 2003 and 24 August 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-25* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-25* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *19 November 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *11/19/03, 6/24/04, 12/22/05, 7/26/06, 3/9/07*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-25 are presented for examination.

Claim Objections

2. Claims 3-5 and 15-16 are objected to because of the following informalities:

- In claims 3-4 and 15-16, the term "processible" should be amended to -- processable -- to correct the spelling error.
- In claim 5, line 3, "load on of" should be amended to – load of –.

3. Appropriate correction is required.

Specification

4. The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

5. Applicant is required to update the Cross-Reference to Related Applications section of the Specification with the appropriate patent numbers for any applications that have been issued.

6. The specification is objected to for potential 101 problem as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required:

- In claim 25, the "computer usable medium" fails to provide proper antecedent basis for the claimed subject matter.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

7. Claims 1-12 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention:

- In claim 1, line 13, it is indefinite whether the term "the program" refers to the "program module" in lines 9-10 are if a new or second program is intended to be introduced.
- In claims 8-10, line 3, the term "a program module" is indefinite because it is unclear whether this refers to the program module of claim 1, lines 9-10 are if a second program module is being introduced.
- Since the scope of the claim is unclear, claims 1 and 8-10 are found to be indefinite.
- Claims 2-7 and 11-12 are also rejected as being dependent upon rejected claim 1.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

8. Claim 25 is directed to non-statutory subject matter. Claim 25 is directed to a "computer usable medium" that lacks antecedent support in the Specification. One of ordinary skill in the art could be able to interpret the plain meaning of a computer usable medium to include a

transmission medium that contains data signals and carrier waves, which are non-statutory subject matter. Thus, claim 25 fails to place the invention squarely within one statutory class of invention. As such, the claim is drawn to a form of energy. Energy is not one of the four categories of invention and therefore this claim(s) is/are not statutory. Energy is not a series of steps or acts and thus is not a process. Energy is not a physical article or object and as such is not a machine or manufacture. Energy is not a combination of substances and therefor not a composition of matter. Therefore, claim 25 is found to be rejected under 35 USC 101 for being directed to non-statutory subject matter.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

9. Claims 1, 13, and 25 are rejected under 35 U.S.C. 102(e) as being anticipated by Applicant's Admitted Prior Art in the Specification (hereinafter AAP).

10. As to claim 1, AAP teaches a task allocation method of allocating a task selectively to a first processor and a second processor in a multiprocessor system, the first processor having a first instruction set and the second processor a second instruction set, and the task corresponding to a program having an execution efficiency, the method comprising:

allocating a task corresponding to a program module described by the first instruction set to the first processor (page 2, lines 11-15 and page 3, lines 9-18, page 1, lines 26-27 through page 2, lines 1-3);

determining whether or not the execution efficiency of the program is improved if a destination allocated for the task is changed from the first processor to the second processor (page 2, lines 7-27 through page 3, lines 1-8, page 1, lines 26-27 through page 2, lines 1-3); and

changing the destination allocated for the task to the second processor if the execution efficiency of the program is improved (page 2, lines 23-27 through page 3, lines 1-8, page 1, lines 26-27 through page 2, lines 1-3).

11. As to claim 13, it is rejected for the same reasons as stated in the rejection of claim 1.

12. As to claim 25, it is rejected for the same reasons as stated in the rejection of claim 1.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person

having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

13. Claims 2-7, 10, 14-19, and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicant's Admitted Prior Art in the Specification (hereinafter AAP) in view of Kubo (US 6,986,139 B1).

14. As to claim 2, AAP teaches further comprising: a first execution time of the program in a case where the task is allocated to the first processor and a second execution time of the program in a case where the task is allocated to the second processor (determining how short the execution times are, etc.) (page 3, lines 1-4); and determining whether the second execution time is shorter than the first execution time, in order to determine whether or not the execution efficiency of the program is improved (page 2, lines 23-27 through page 3, lines 1-8).

15. AAP is silent in estimating the performance/execution times. However, Kubo teaches a task allocation multiprocessor system that estimates the performance/load indexes of each processor in order to make a decision on whether another processor would be better to process the task (Abstract, col. 4, lines 12-48). AAP and Kubo are analogous art because they are both in the same field of endeavor of task allocation management. One of ordinary skill in the art would have known to modify AAP such that it would estimate performance to determine if another processor should be allocated the task. The suggestion/motivation for doing so would have been to provide the predicted result of having an effective and low-overhead way to load balance the resources (col. 4, lines 3-11). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Kubo to obtain the invention of claim 2.

16. As to claim 3, AAP teaches further comprising: a first data amount processible by the task within a unit time in a case where the task is allocated to the first processor and a second data amount processible by the task within a unit time in a case where the task is allocated to the second processor (determining how large the process data amount per unit time is, etc.) (page 3, lines 1-5); and determining whether the second data amount is larger than the first data amount, in order to determine whether or not the execution efficiency of the program is improved (page 2, lines 23-27 through page 3, lines 1-8).

17. AAP is silent in estimating the performance/data amount. However, Kubo teaches a task allocation multiprocessor system that estimates the performance/load indexes of each processor in order to make a decision on whether another processor would be better to process the task (Abstract, col. 4, lines 12-48). AAP and Kubo are analogous art because they are both in the same field of endeavor of task allocation management. One of ordinary skill in the art would have known to modify AAP such that it would estimate performance to determine if another processor should be allocated the task. The suggestion/motivation for doing so would have been to provide the predicted result of having an effective and low-overhead way to load balance the resources (col. 4, lines 3-11). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Kubo to obtain the invention of claim 3.

18. As to claim 4, AAP teaches further comprising: a first data amount processible by the task within a unit time in a case where the task is allocated to the first processor and estimating a

second data amount processible by the task within a unit time in a case where the task is allocated to the second processor; an increment of data amount between the first data amount and the second data amount (determining how large the process data amount per unit time is, etc.) (page 3, lines 1-5); and determining whether the increment is larger than a preset threshold, in order to determine whether or not the execution efficiency of the program is improved (page 2, lines 23-27 through page 3, lines 1-8).

19. AAP is silent in estimating the performance/increment of data amount. However, Kubo teaches a task allocation multiprocessor system that estimates the performance/load indexes of each processor in order to make a decision on whether another processor would be better to process the task (Abstract, col. 4, lines 12-48). AAP and Kubo are analogous art because they are both in the same field of endeavor of task allocation management. One of ordinary skill in the art would have known to modify AAP such that it would estimate performance to determine if another processor should be allocated the task. The suggestion/motivation for doing so would have been to provide the predicted result of having an effective and low-overhead way to load balance the resources (col. 4, lines 3-11). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Kubo to obtain the invention of claim 4.

20. As to claim 5, AAP teaches further comprising: a load on of the second processor in a case where the destination allocated for the task is changed from the first processor to the second processor (determining how small the load on each processor is, etc.) (page 3, lines 5-6); determining whether the load on the second processor is an overload, in order to determine

whether or not the execution efficiency of the program is improved (page 2, lines 23-27 through page 3, lines 1-8).

21. AAP is silent in estimating the performance/data amount. However, Kubo teaches a task allocation multiprocessor system that estimates the performance/load indexes of each processor in order to make a decision on whether another processor would be better to process the task (Abstract, col. 4, lines 12-48). AAP and Kubo are analogous art because they are both in the same field of endeavor of task allocation management. One of ordinary skill in the art would have known to modify AAP such that it would estimate performance to determine if another processor should be allocated the task. The suggestion/motivation for doing so would have been to provide the predicted result of having an effective and low-overhead way to load balance the resources (col. 4, lines 3-11). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Kubo to obtain the invention of claim 5.

22. As to claim 6, AAP teaches further comprising: a first amount of data transferred by inter-processor communication in the program in a case where the task is allocated to the first processor and a second amount of data transferred by inter-processor communication (inter-processor connection unit such as a bus or a crossbar switch) in the program in a case where the task is allocated to the second processor (determining how small the load on each processor is, etc.) (page 3, lines 5-6, page 1, lines 26-27 through page 2, line 1); and determining whether the second amount of data is smaller than the first amount of data, in order to determine whether or

not the execution efficiency of the program is improved (page 2, lines 23-27 through page 3, lines 1-8).

23. AAP is silent in estimating the performance/data amount. However, Kubo teaches a task allocation multiprocessor system that estimates the performance/load indexes of each processor in order to make a decision on whether another processor would be better to process the task (Abstract, col. 4, lines 12-48). AAP and Kubo are analogous art because they are both in the same field of endeavor of task allocation management. One of ordinary skill in the art would have known to modify AAP such that it would estimate performance to determine if another processor should be allocated the task. The suggestion/motivation for doing so would have been to provide the predicted result of having an effective and low-overhead way to load balance the resources (col. 4, lines 3-11). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Kubo to obtain the invention of claim 6.

24. As to claim 7, AAP teaches further comprising: a first amount of data transferred by inter-processor communication within a unit time in the program in a case where the task is allocated to the first processor and a second amount of data transferred by inter-processor communication (inter-processor connection unit such as a bus or a crossbar switch) within a unit time in the program in a case where the task is allocated to the second processor (determining how small the load on each processor is, etc.) (page 3, lines 5-6, page 1, lines 26-27 through page 2, line 1); and determining whether the second amount of data within the unit time is smaller

than the first amount of data within the unit time, in order to determine whether or not the execution efficiency of the program is improved (page 2, lines 23-27 through page 3, lines 1-8).

25. AAP is silent in estimating the performance/data amount. However, Kubo teaches a task allocation multiprocessor system that estimates the performance/load indexes of each processor in order to make a decision on whether another processor would be better to process the task (Abstract, col. 4, lines 12-48). AAP and Kubo are analogous art because they are both in the same field of endeavor of task allocation management. One of ordinary skill in the art would have known to modify AAP such that it would estimate performance to determine if another processor should be allocated the task. The suggestion/motivation for doing so would have been to provide the predicted result of having an effective and low-overhead way to load balance the resources (col. 4, lines 3-11). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Kubo to obtain the invention of claim 7.

26. As to claim 10, AAP teaches further comprising: acquiring a program module described by the second instruction set (page 2, lines 7-19). AAP is silent in teaching the use of a file system or a network. However, Kubo teaches a task allocation multiprocessor system that uses a communication network 51 as a means to communicate data such as estimations of performance/load indexes of each processor in order to make a decision on whether another processor would be better to process the task (Abstract, col. 4, lines 12-48, Fig. 1, item 51). AAP and Kubo are analogous art because they are both in the same field of endeavor of task allocation management. One of ordinary skill in the art would have known to modify AAP such

that it would include the communication network as taught in Kubo. The suggestion/motivation for doing so would have been to provide the predicted result of having a means to communicate important information/data that would create an effective and low-overhead way to load balance the resources (col. 4, lines 3-11). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Kubo to obtain the invention of claim 10.

27. As to claims 14-19, they are rejected for the same reasons as stated in the rejections of claims 2-7, respectively.

28. As to claim 22 it is rejected for the same reasons as stated in the rejection of claim 10.

29. Claims 8-9, 11, 20-21, and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicant's Admitted Prior Art in the Specification (hereinafter AAP) in view of Gschwind et al. (hereinafter Gschwind) (US 2004/0083462 A1).

30. As to claim 8, AAP teaches a multiprocessor system wherein each processor has instruction sets and a program module where tasks can be allocated to a different processor based on performance values such as load, execution time, etc. (see rejection of claim 1). AAP is silent

in acquiring a program module described by the second instruction set, which is necessary for creating a task to be allocated to the second processor, by replacing a first instruction, in the program module described by the first instruction set, with a second instruction of the second instruction set, for executing the same process as the first instruction.

31. However, Gschwind discloses a multiprocessor system (Fig. 1, items 120, 130) with code/instruction sets partitioned into a plurality of modules, wherein the modules comprise at least one main processor code module and at least one attached processor code module. Code is selected to be encapsulated within at least one main processor code module, wherein the selected code is targeted for transfer and employment by at least one attached processor. At least one main processor code module and at least one selected attached processor code module are compiled. The compiled processor code targeted for the main processing module is linked to a stub function and a linked object integrated executable program is created (page 1, [0008], page 2, [0018]-[0019]). AAP and Gschwind are analogous art because they are both in the same field of endeavor of multiprocessor execution. One of ordinary skill in the art would have known to modify AAP such that it would include the transferring and compiling feature as mentioned above and as taught in Gschwind. The suggestion/motivation for doing so would have been to provide the predicted result of a standardization of passing information between a plurality of processors, a means to create an integrated executable, in addition to increasing efficiencies in the programming process (page 1, [0002], last 7 lines of [0006] and [0007], page 2, lines 1-3 of [0022]). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Gschwind to obtain the invention of claim 8.

32. As to claim 9, AAP teaches further comprising: acquiring a program module described by the second instruction set, which is necessary for creating a task to be allocated to the second processor (see rejection of claim 1). AAP is silent in compiling a source code for the program module described by the first instruction set with a compiler for the second processor.

33. However, Gschwind discloses a multiprocessor system (Fig. 1, items 120, 130) with code/instruction sets partitioned into a plurality of modules, wherein the modules comprise at least one main processor code module and at least one attached processor code module. Code is selected to be encapsulated within at least one main processor code module, wherein the selected code is targeted for employment by at least one attached processor. At least one main processor code module and at least one selected attached processor code module are compiled. The compiled processor code targeted for the main processing module is linked to a stub function and a linked object integrated executable program is created (page 1, [0008]). AAP and Gschwind are analogous art because they are both in the same field of endeavor of multiprocessor execution. One of ordinary skill in the art would have known to modify AAP such that it would include the compiling feature as mentioned above and as taught in Gschwind. The suggestion/motivation for doing so would have been to provide the predicted result of a standardization of passing information between a plurality of processors, a means to create an integrated executable, in addition to increasing efficiencies in the programming process (page 1, [0002], last 7 lines of [0006] and [0007], page 2, lines 1-3 of [0022]). Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Gschwind to obtain the invention of claim 9.

34. As to claim 11, AAP teaches the use of a program module (see rejection of claim 1). However, AAP is explicitly silent wherein the program module is a program module complex including a first program module described by the first instruction set for the first processor and a second program module described by the second instruction set for the second processor, and wherein said task is created by using one of the first program module and the second program module.

35. However, Gschwind discloses a multiprocessor system (Fig. 1, items 120, 130) with code/instruction sets partitioned into a plurality of modules, wherein the modules comprise at least one main processor code module and at least one attached processor code module. Code is selected to be encapsulated within at least one main processor code module, wherein the selected code is targeted for employment by at least one attached processor. At least one main processor code module and at least one selected attached processor code module are compiled. The compiled processor code targeted for the main processing module is linked to a stub function and a linked object integrated executable program is created (page 1, [0008]). AAP and Gschwind are analogous art because they are both in the same field of endeavor of multiprocessor execution. One of ordinary skill in the art would have known to modify AAP such that it would include the complex program module (attached processor code module and main processor code module) and compiler feature as mentioned above and as taught in Gschwind. The suggestion/motivation for doing so would have been to provide the predicted result of a standardization of passing information between a plurality of processors in addition to increasing efficiencies in the programming process (page 1, [0002], last 7 lines of [0006] and [0007]).

Therefore, it would have been obvious to one of ordinary skill in the art to combine AAP and Gschwind to obtain the invention of claim 11.

36. As to claims 20-21, they are rejected for the same reasons as stated in the rejections of claims 8-9, respectively.

37. As to claim 23 it is rejected for the same reasons as stated in the rejection of claim 11.

38. Claims 12 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicant's Admitted Prior Art in the Specification (hereinafter AAP) in view of Yokoya (US 6,199,093 B1).

39. As to claim 12, AAP teaches changing of the destination allocated for the task to the second processor (see rejection of claim 1). AAP is silent in teaching updating a task allocation table storing task allocation information. However, Yokoya teaches task allocation in a multiprocessor system such that a task allocation table is used to store task allocation information (see Abstract, Fig. 1, items 4, 6, 7, 8, 9, col. 3, lines 40-67, etc.). AAP and Yokoya are analogous art because they are both in the same field of endeavor of task allocation and both

solving the same problem as optimizing the task allocation. One of ordinary skill in the art would have known to modify AAP such that it would include the task allocation used to update and store task allocation information. The suggestion/motivation for doing so would have been to provide the predicted result of providing data communication in a fast, efficient, and automatic way (col. 2, lines 5-19). Therefore, it would have been obvious to combine AAP and Yokoya to obtain the invention of claim 12.

40. As to claim 24 it is rejected for the same reasons as stated in the rejection of claim 12.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure:

- **Smith (US 5,694,602)** discloses a multiprocessor system and method of allocating loads and estimating weights for the plurality of processors so that load balancing could be performed based on the estimation (see Abstract).
- **Klarer et al. (US 7,213,238 B2)** discloses acquiring a program module described by the second instruction set, which is necessary for creating a task to be allocated to the second processor, by compiling a source code for the program module described by the first instruction set with a compiler for the second processor (see claim 1 and Abstract).

- **Debenedictis et al. (US 5,625,823)** discloses the use of a compiler that generates an instruction set independent output. The method changes the Inter-task representation by compiling the source code to the instruction-set independent output, replacing the source code in the tasks with this output (col. 21, lines 39-46).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to KENNETH TANG whose telephone number is (571)272-3772. The examiner can normally be reached on 8:30AM - 6:00PM, Every other Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Meng-Ai An/
Supervisory Patent Examiner, Art Unit 2195

/Kenneth Tang/
Examiner, Art Unit 2195